IN THE CLAIMS

1. (Canceled)

2. (Previously Presented) A telecommunications system, comprising:

a plurality of framer farms adapted to generate a corresponding plurality of event signals responsive to a corresponding plurality of signaling events, where the event signals are to request service for corresponding framer farms;

an event manager adapted to sequence the plurality of event signals; and

wherein each framer farm comprises

a plurality of framers adapted to receive a corresponding plurality of digital signals, extract therefrom corresponding signaling data, and generate a corresponding event signal responsive to the corresponding signaling data;

a signaling queue adapted to queue the corresponding signaling data; and

a signal register adapted to read the corresponding signaling queue.

3. (Original) The system of claim 2 wherein the signaling queue is a circular queue.

4. (Previously Presented) The system of claim 3 wherein the signaling queue includes a signaling read and write pointer, wherein reading the signal register causes the corresponding signaling data pointed to by the read pointer to appear in the signal register and increments the signaling read pointer.

5. (Previously Presented) The system of claim 2 wherein the signal register comprises:

a signaling data field adapted to store signaling data for a predetermined framer;

a last entry field adapted to identify a last entry for the predetermined framer;

a framer identification field adapted to identify the predetermined framer; and

a timeslot identification field adapted to identify a timeslot related to the signaling data for the predetermined framer.

6-9. (Canceled)

10. (Previously Presented) A telecommunications system, comprising:

a plurality of framer farms adapted to generate a corresponding plurality of event signals responsive to a corresponding plurality of signaling events, where the event signals are to request service for corresponding framer farms;

an event manager adapted to sequence the plurality of event signals;

wherein the event manager comprises:

an event queue adapted to queue the plurality of event signals,

a status register adapted to maintain a status of the event queue, and

an event register adapted to read the event queue; and

wherein the event queue includes an event read and write pointers, wherein reading the event register causes the event signal pointed to by the event read pointer to appear in the event register and increments the event read pointer.

11. (Previously Presented) A telecommunications system, comprising:

a plurality of framer farms adapted to generate a corresponding plurality of event signals responsive to a corresponding plurality of signaling events, where the event signals are to request service for corresponding framer farms;

an event manager adapted to sequence the plurality of event signals;

wherein the event manager comprises:

an event queue adapted to queue the plurality of event signals,

a status register adapted to maintain a status of the event queue, and

an event register adapted to read the event queue; and

wherein the event manager comprises:

a queue depth register adapted to indicate a number of event signals received at the event queue; and

a maximum queue depth register adapted to indicate a maximum number of event signals capable of being received at the event queue.

12. (Original) The system of claim 11 wherein the event manager comprises a partially full register adapted to indicate a fraction of the maximum number of event signals capable of being received at the event queue.

13. (Original) The system of claim 12 wherein the partially full register comprises:

a partially full high register adapted to indicate a high threshold of the maximum number of event signals capable of being received at the event queue; and

a partially full low register adapted to indicate a low threshold of the maximum number of event signals capable of being received at the event queue;

wherein the partially full high and low registers provide the event queue with hysteresis.

14. (Previously Presented) A telecommunications system, comprising:

a plurality of framer farms adapted to generate a corresponding plurality of event signals responsive to a corresponding plurality of signaling events, where the event signals are to request service for corresponding framer farms;

an event manager adapted to sequence the plurality of event signals;

wherein the event manager comprises:

an event queue adapted to queue the plurality of event signals,

a status register adapted to maintain a status of the event queue, and

an event register adapted to read the event queue; and

wherein the status register comprises:

a queue full field adapted to indicate when the event queue is full;

an almost full field adapted to indicate when the event queue is almost full;

an overrun field adapted to indicate when signaling events where missed due to the event queue being full; and

a queue empty field adapted to indicate when the signaling queue is empty.

15-18. (Canceled)

19. (Previously Presented) An event manager, comprising:

an event queue adapted to queue a plurality of event signals;

a status register adapted to maintain a status of each of the plurality of event signals;

an event register adapted to read the event queue;

a queue depth register adapted to indicate a number of event signals received at the event queue;

a maximum queue depth register adapted to indicate a maximum number of event signals capable of being received at the event queue; and

a partially full register adapted to indicate a fraction of the maximum number of event signals capable of being received at the event queue.

20. (Original) The system of claim 19 wherein the partially full register comprises:

a partially full high register adapted to indicate a high threshold of the maximum number of event signals capable of being received at the event queue; and

a partially full low register adapted to indicate a low threshold of the maximum number of event signals capable of being received at the event queue;

wherein the partially full high and low registers provide the event queue with hysteresis.

21. (Original) The system of claim 20 wherein the status register comprises:

a queue full field adapted to indicate when the event queue is full;

an almost full field adapted to indicate when the event queue is almost full;

an overrun field adapted to indicate when signaling events where missed due to the event queue being full; and

a queue empty field adapted to indicate when the signaling queue is empty.

22-25. (Canceled)

26. (Previously Presented) An event manager, comprising:

an event queuing means adapted to queue a plurality of event signals;

a status registering means adapted to maintain a status of each of the event signals; and

an event registering means adapted to read the event queue;

a queue depth registering means adapted to indicate a number of event signals received at the event queuing means;

a maximum queue depth registering means adapted to indicate a maximum number of event signals capable of being received at the event queuing means; and

a partially full registering means adapted to indicate a fraction of the maximum number of event signals capable of being received at the event queuing means.

27. (Original) The system of claim 26 wherein the partially full registering means comprises:

a partially full high registering means adapted to indicate a high threshold of the maximum number of event signals capable of being received at the event queuing means; and

a partially full low registering means adapted to indicate a low threshold of the maximum number of event signals capable of being received at the event queuing means;

wherein the partially full high and low registers provide the event queue with hysteresis.

28. (Original) The system of claim 27 wherein the status registering means comprises:

a queue full means adapted to indicate when the event queue is full;

an almost full means adapted to indicate when the event queue is almost full;

an overrun means adapted to indicate when signaling events where missed due to the event queue being full; and

a queue empty means adapted to indicate when the signaling queue is empty.

29. (Original) A method for processing signaling bit changes for telecommunication transmitted over a global network, comprising:

receiving a plurality of digital signals in a framer farm, the framer farm including a plurality of framers corresponding the plurality of digital signals, each digital signal including corresponding signaling bits;

extracting the signaling bits for each framer;

queuing the signaling bits in a signaling queue;

generating an event signal responsive to the queuing;

queuing the event signal in an event queue; and

processing the signaling bits responsive to the event signal.

30. (Original) The method of claim 29 wherein queuing the signaling bits includes:

associating the signaling bits with an originating framer;

storing the signaling bits for the originating framer;

identifying a last entry for the originating framer; and

identifying a timeslot related to the signaling bits for the originating framer.

31. (Original) The method of claim 29 wherein queuing the signaling bits includes reading the signaling queue with a signaling register.

32. (Original) The method of claim 31 wherein queuing the signaling bits includes circularly reading the signaling queue.

33. (Original) The method of claim 32 wherein circularly reading the signaling queue includes reading the signaling register and incrementing a signaling read pointer.

34. (Original) The method of claim 29 wherein generating an event signal includes generating a single event signal for every framer farm having framers issuing signaling bits.

35. (Original) The method of claim 29 wherein queuing the event signal includes sequentially storing the event signals.

36. (Original) The method of claim 29 wherein queuing the event signal includes reading the event queue with an event register.

37. (Original) The method of claim 29 wherein processing the signaling bits includes incrementing a queue depth register every time a new event signal is queued in the event queue and decrementing the queue depth register every time a queued event signal is removed from the event queue.

38. (Original) The method of claim 37 wherein processing the signaling bits includes comparing the queue depth register with a maximum queue depth register and setting a queue full bit if the queue depth register is greater than or equal to the maximum queue depth register.

39. (Original) The method of claim 38 wherein processing the signaling bits includes setting an overrun bit if event signals are not queued because the event queue is full.

40. (Original) The method of claim 38 wherein processing the signaling bits includes programming a partially full register to indicate a fraction of the maximum queue depth register with a high and a low threshold.

41. (Original) The method of claim 40 wherein processing the signaling bits includes setting an almost full bit when the queue depth register is greater than or equal to the partially full register.

42. (Original) A computer readable medium having stored thereon instructions, that, when executed by a computing device, result in:

receiving a plurality of digital signals in a framer farm, the framer farm including a plurality of framers corresponding the plurality of digital signals, each digital signal including corresponding signaling bits;

extracting the signaling bits for each framer;

queuing the signaling bits in a signaling queue;

generating an event signal responsive to the queuing;

queuing the event signal in an event queue; and

processing the signaling bits responsive to the event signal.

43. (Original) The method of claim 42 wherein queuing the signaling bits includes:

associating the signaling bits with an originating framer;

storing the signaling bits for the originating framer;

identifying a last entry for the originating framer; and

identifying a timeslot related to the signaling bits for the originating framer.

44. (Original) The method of claim 42 wherein queuing the signaling bits includes reading the signaling queue with a signaling register.

45. (Original) The method of claim 44 wherein queuing the signaling bits includes circularly reading the signaling queue.

46. (Original) The method of claim 45 wherein circularly reading the signaling queue includes reading the signaling register and incrementing a signaling read pointer.

47. (Original) The method of claim 42 wherein generating an event signal includes generating a single event signal for every framer farm having framers issuing signaling bits.

48. (Original) The method of claim 42 wherein queuing the event signal includes sequentially storing the event signals.

49. (Original) The method of claim 42 wherein queuing the event signal includes reading the event queue with an event register.

50. (Original) The method of claim 42 wherein processing the signaling bits includes incrementing a queue depth register every time a new event signal is queued in the event queue and decrementing the queue depth register every time a queued event signal is removed from the event queue.

51. (Original) The method of claim 50 wherein processing the signaling bits includes comparing the queue depth register with a maximum queue depth register and setting a queue full bit if the queue depth register is greater than or equal to the maximum queue depth register.

52. (Original) The method of claim 51 wherein processing the signaling bits includes setting an overrun bit if event signals are not queued because the event queue is full.

53. (Original) The method of claim 51 wherein processing the signaling bits includes programming a partially full register to indicate a fraction of the maximum queue depth register with a high and a low threshold.

54. (Original) The method of claim 53 wherein processing the signaling bits includes setting an almost full bit when the queue depth register is greater than or equal to the partially full register.